#4. Zadatak

```
library(readr)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

Loading dataset

```
dataset <- read_csv("preprocessed_data.csv")
```

```
## Rows: 1460 Columns: 81
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (43): MSZoning, Street, Alley, LotShape, LandContour, Utilities, LotConf...
## dbl (38): Id, MSSubClass, LotFrontage, LotArea, OverallQual, OverallCond, Ye...
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
not_normal_neighbourhoods <- c("Blmngtn", "Blueste", "BrDale", "ClearCr","IDOTRR", "MeadowV", "Mitchel"
dataset <- subset(dataset, !Neighborhood %in% not_normal_neighbourhoods)
```

Višestruka regresija

Prije procjene modela višestruke regresije potrebno je provjeriti da pojedini parovi varijabli nisu (previše) korelirani. U principu je određena korelacija između varijabli neizbježna, ali varijable s vrlo visokom korelacijom će uzrokovati probleme u interpretaciji regresijskih rezultata.

Regresija s jako koreliranim ulaznim varijablama će uglavnom dati neke rezultate, ali na temelju njih ne možemo donositi nikakve zaključke. U slučaju savršene linearne zavisnosti ili koreliranosti ulaznih varijabli, procjena regresijskog modela će biti nestabilna i barem jedan koeficijent će biti NA.

Stoga je potrebo odabrati onaj podskup varijabli za koje smatramo da objašnjavaju različite efekte u podatcima i nisu međusobno (previše) korelirane.

```
neighborhood_factor <- factor(dataset$Neighborhood)
neighborhood_numeric <- as.numeric(neighborhood_factor)
table(neighborhood_numeric)
```

```
## neighborhood_numeric
##   1   2   3   4   5   6   7   8   9  10  11
##  58 150  51 100  79 225  77  73 113  74  59
```

```
cor(dataset$YearBuilt, dataset$SalePrice)
```

```
## [1] 0.5163603
```

```r
cor(dataset$GrLivArea, dataset$SalePrice)
```

```
## [1] 0.6673089
```

```r
cor(dataset$OverallQual, dataset$SalePrice)
```

```
## [1] 0.795998
```

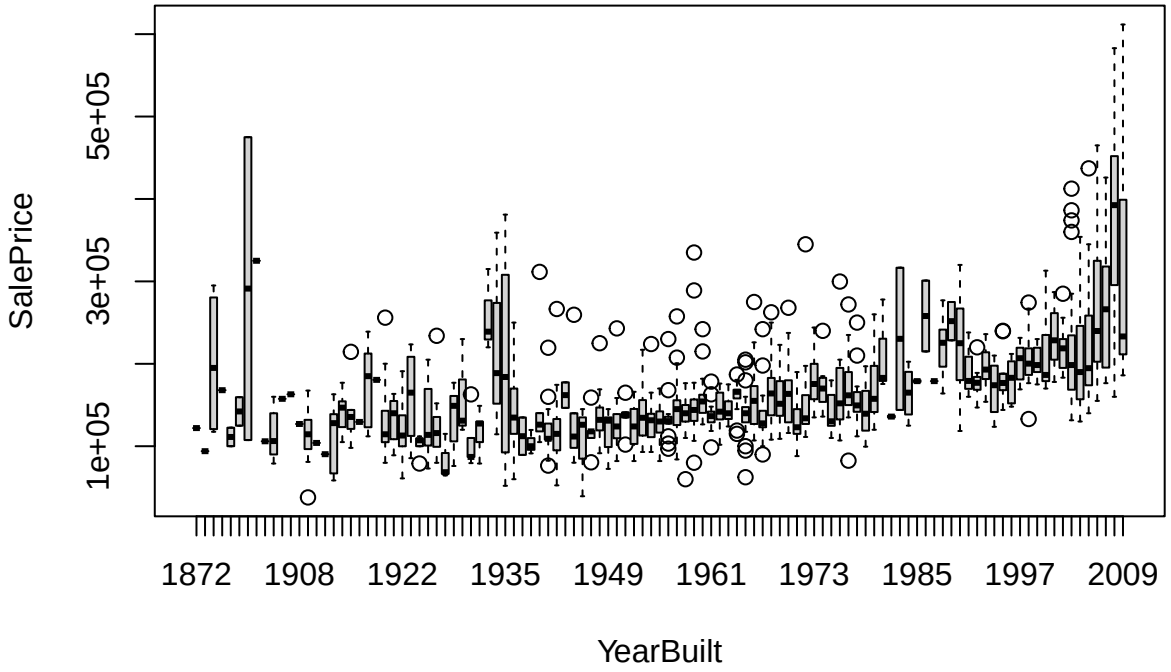
```r
cor(neighborhood_numeric, dataset$SalePrice)
```

```
## [1] -0.03752398
```

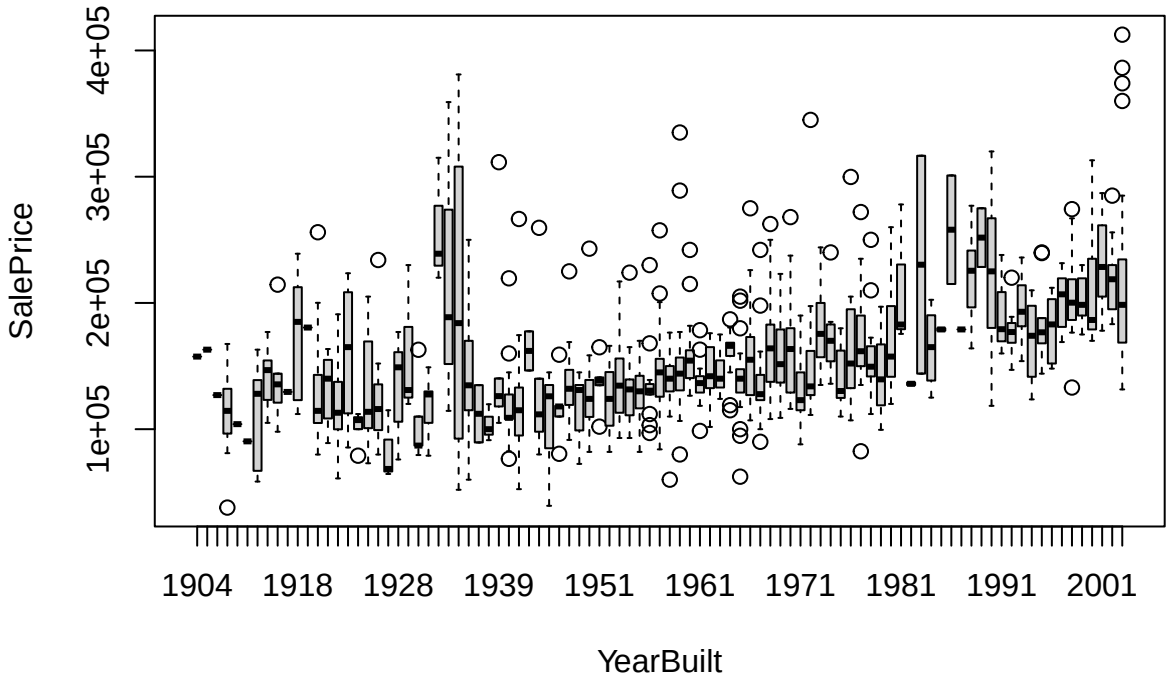
```r
cor(cbind(dataset$YearBuilt, dataset$GrLivArea, dataset$OverallQual, neighborhood_numeric)) # korelacij
```

```
##                                                  neighborhood_numeric
##                      1.0000000 0.17717588  0.51859655          -0.10639943
##                      0.1771759 1.00000000  0.60403848           0.04702938
##                      0.5185965 0.60403848  1.00000000          -0.04900663
## neighborhood_numeric -0.1063994 0.04702938 -0.04900663           1.00000000
```

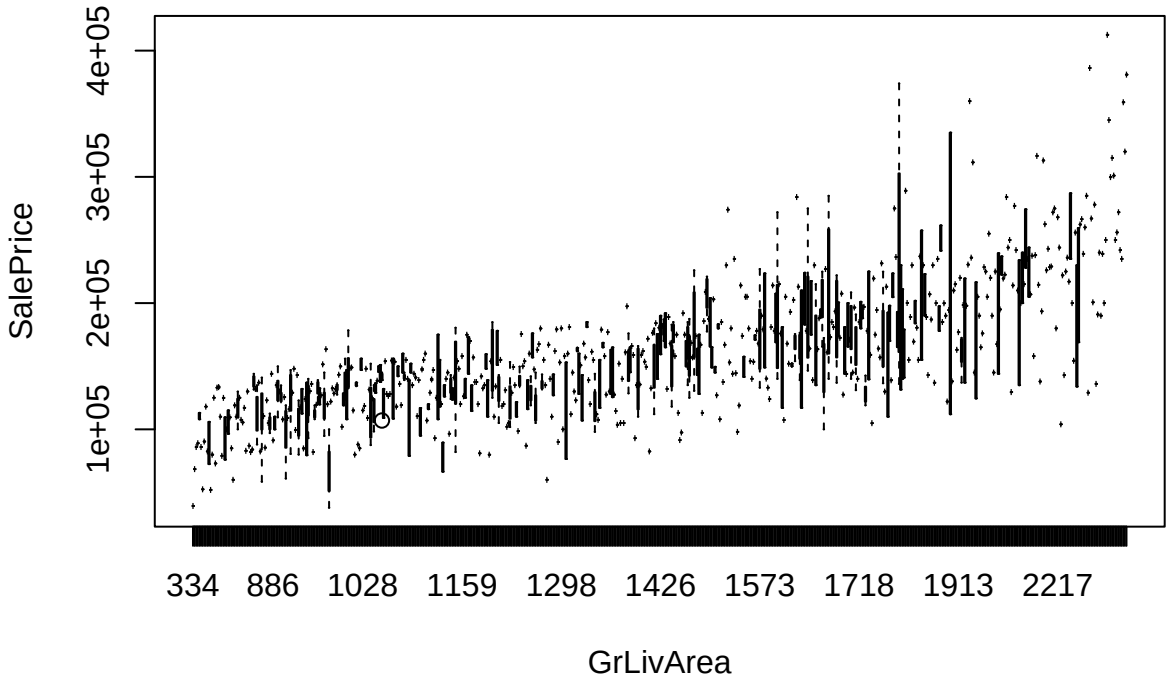
```r
boxplot(SalePrice ~ YearBuilt , data = dataset) #kvadratni dijagram se moze koristiti za graficki provj
```



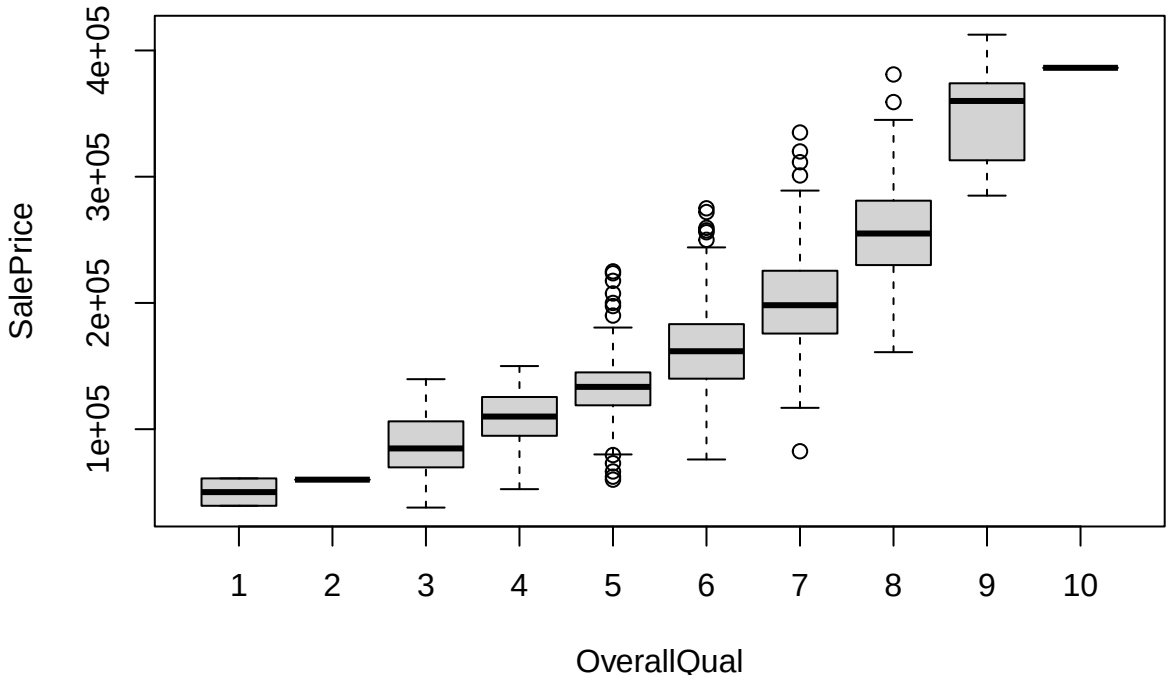

```r
dataset <- dataset %>%
  group_by(YearBuilt) %>%
  filter(YearBuilt < 2004 & YearBuilt > 1900)
boxplot(SalePrice ~ YearBuilt , data = dataset)
```



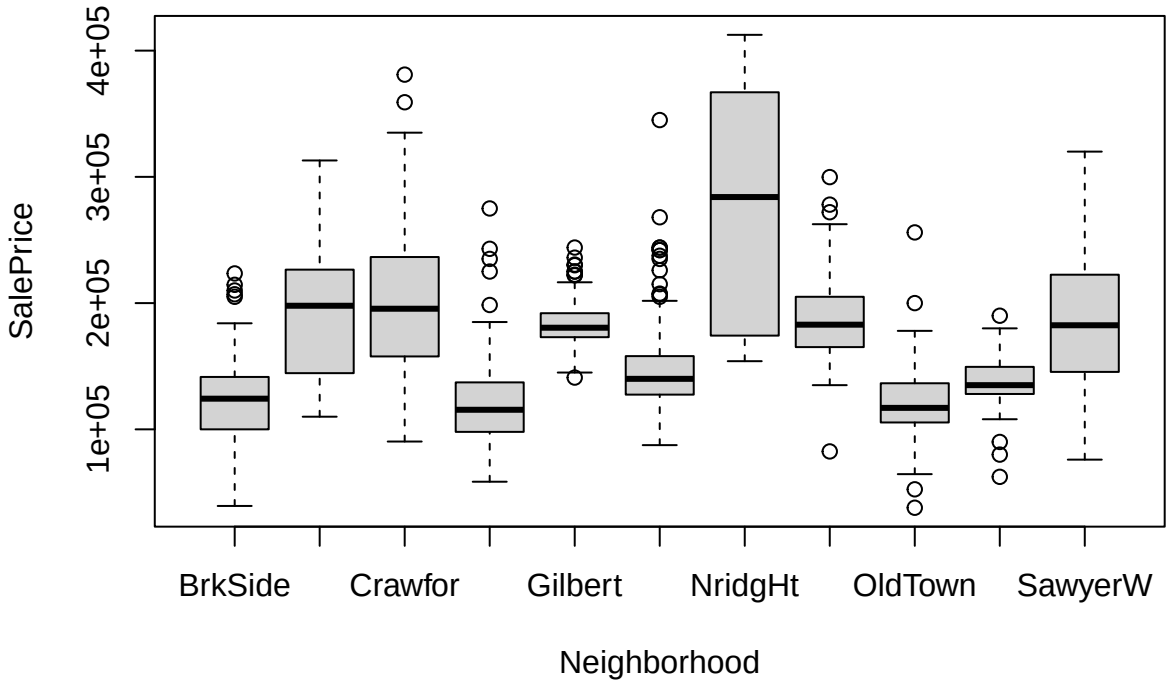

```
boxplot(SalePrice ~ GrLivArea , data = dataset)
```




```
boxplot(SalePrice ~ OverallQual , data = dataset)
```

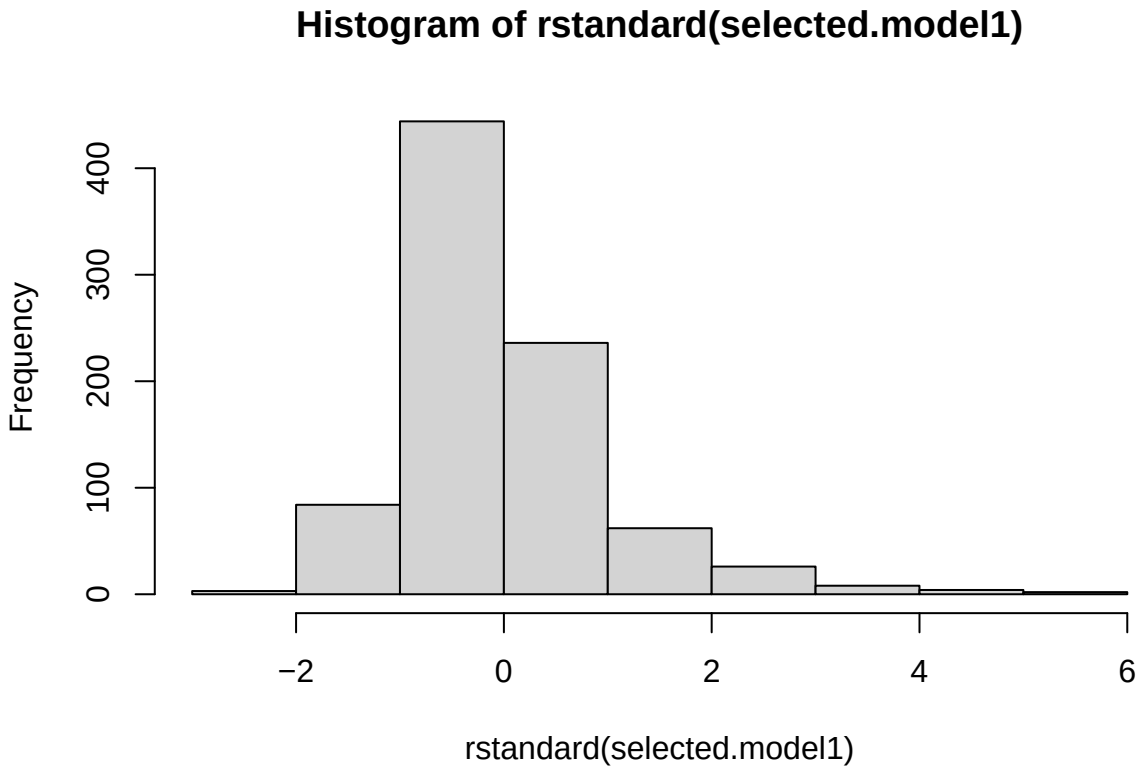



```
boxplot(SalePrice ~ Neighborhood , data = dataset)
```

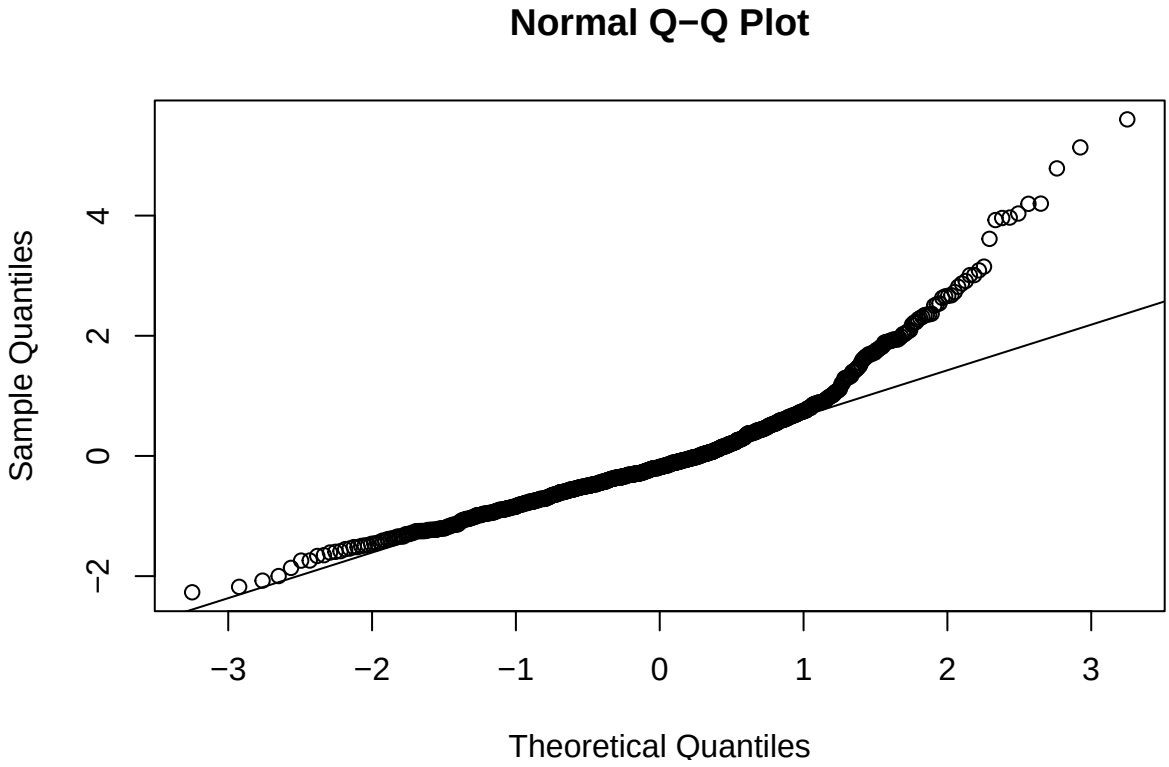



```
fit.YearBuilt = lm(SalePrice ~ YearBuilt , data=dataset)
fit.GrLivArea = lm(SalePrice ~ GrLivArea , data=dataset)
fit.OverallQual = lm(SalePrice ~ OverallQual , data=dataset)
fit.Neighborhood = lm(SalePrice ~ Neighborhood , data=dataset)
```

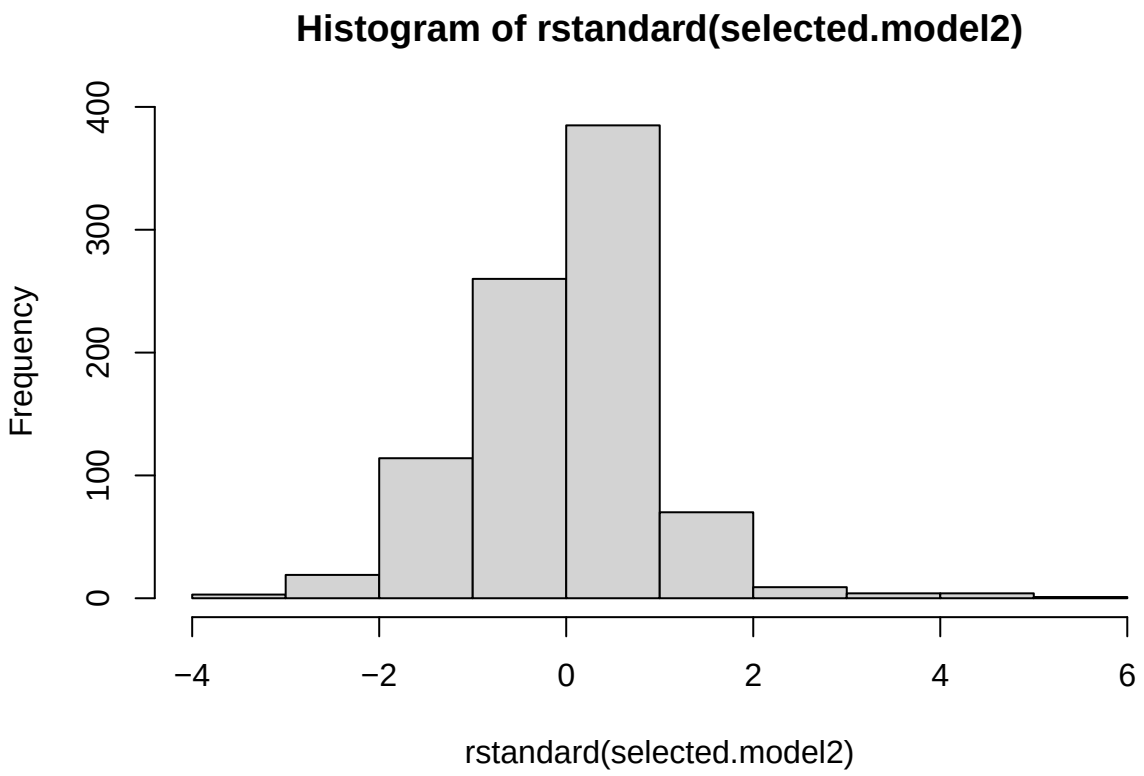
```
selected.model1 = fit.YearBuilt
hist(rstandard(selected.model1))
```



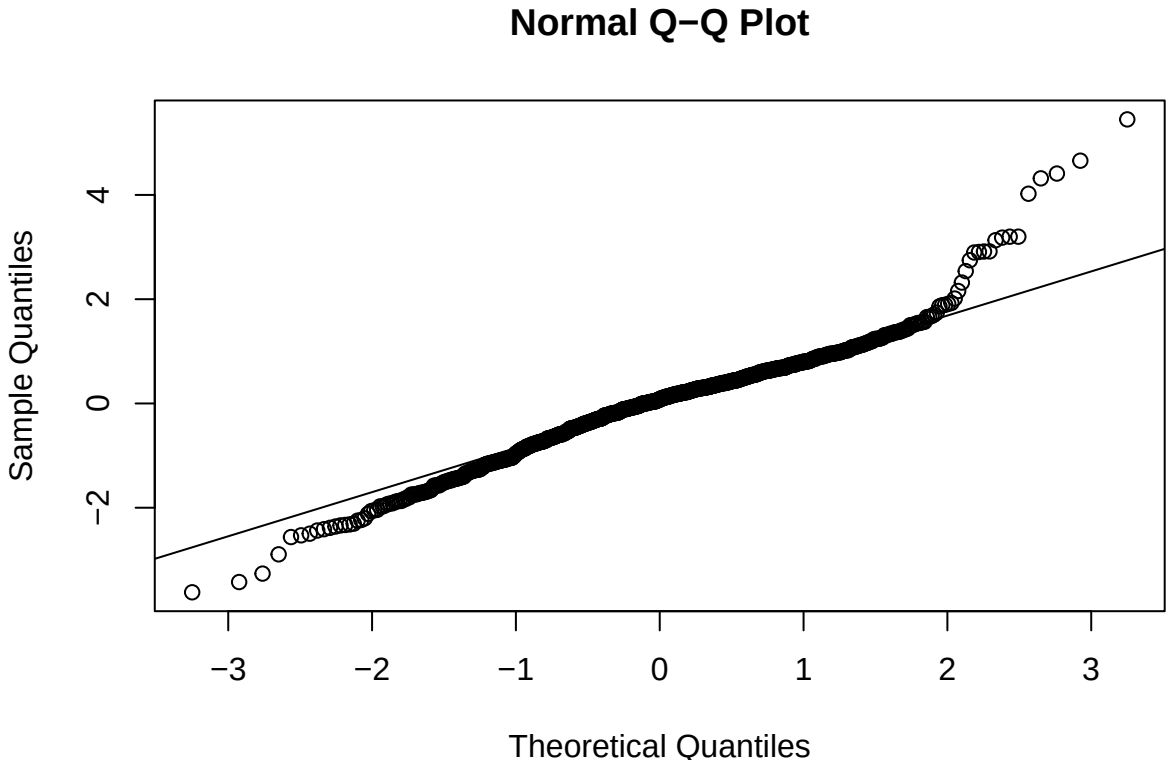

```
qqnorm(rstandard(selected.model1))
qqline(rstandard(selected.model1))
```



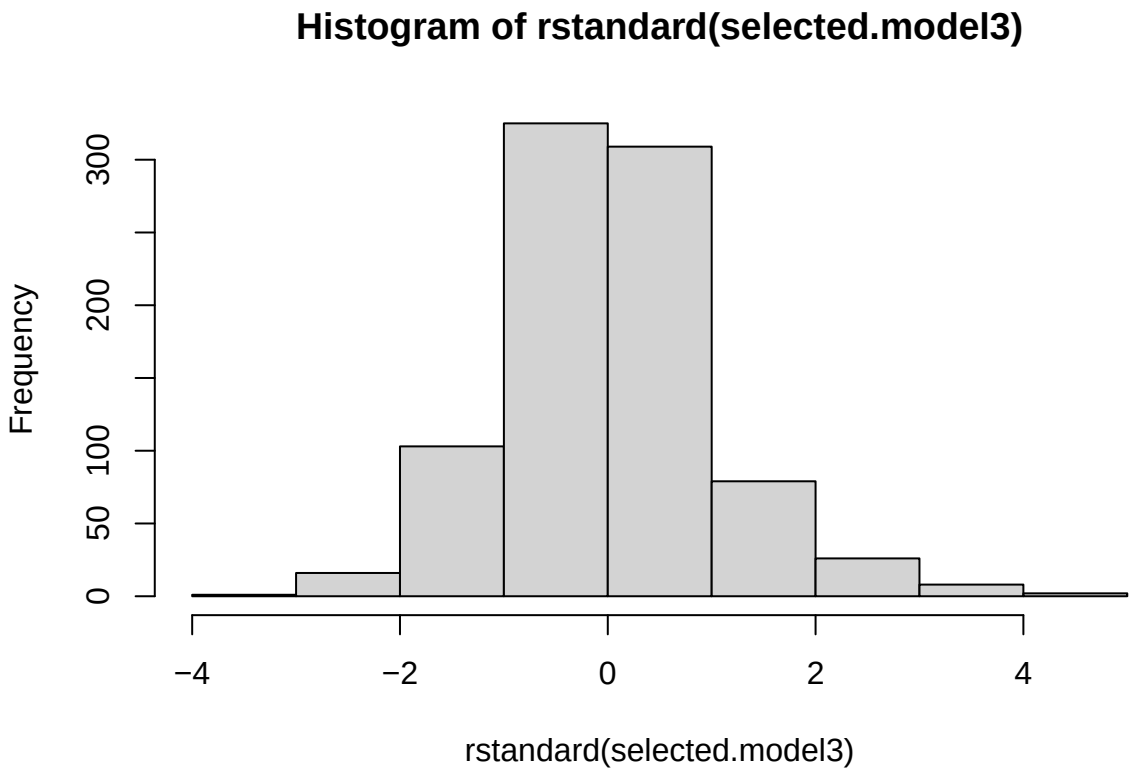

```
selected.model2 = fit.GrLivArea
hist(rstandard(selected.model2))
```



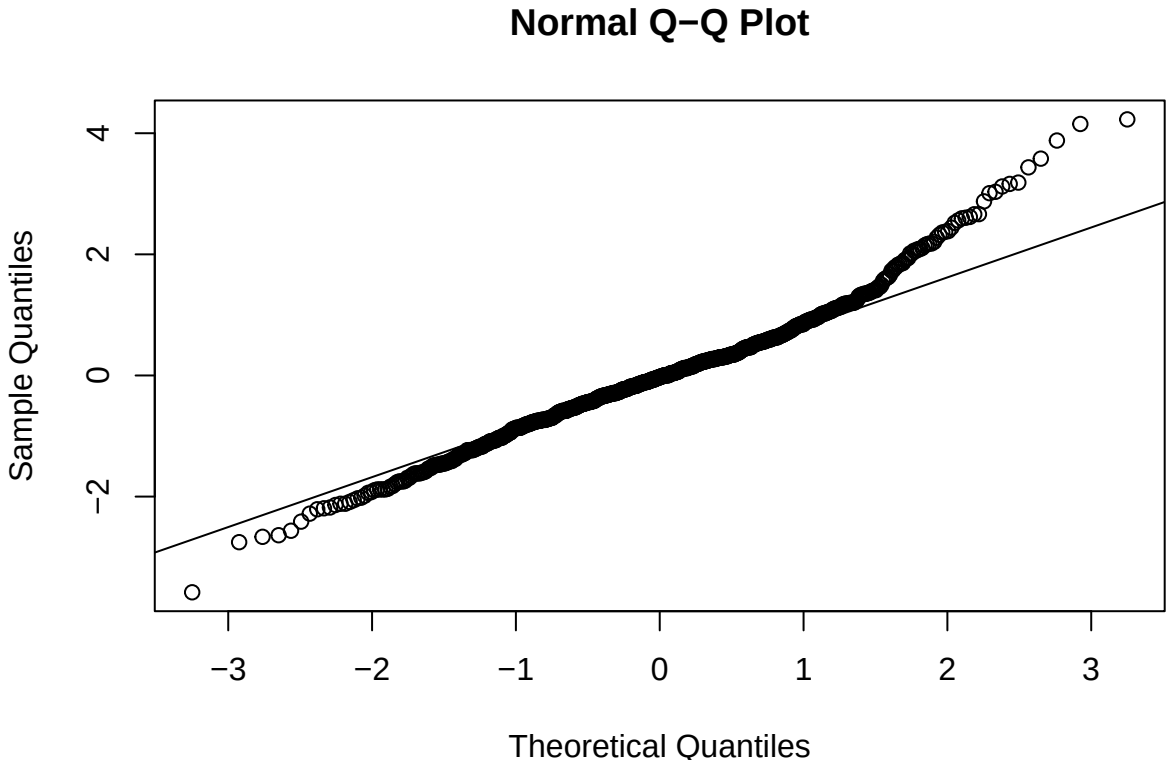

```
qqnorm(rstandard(selected.model2))
qqline(rstandard(selected.model2))
```



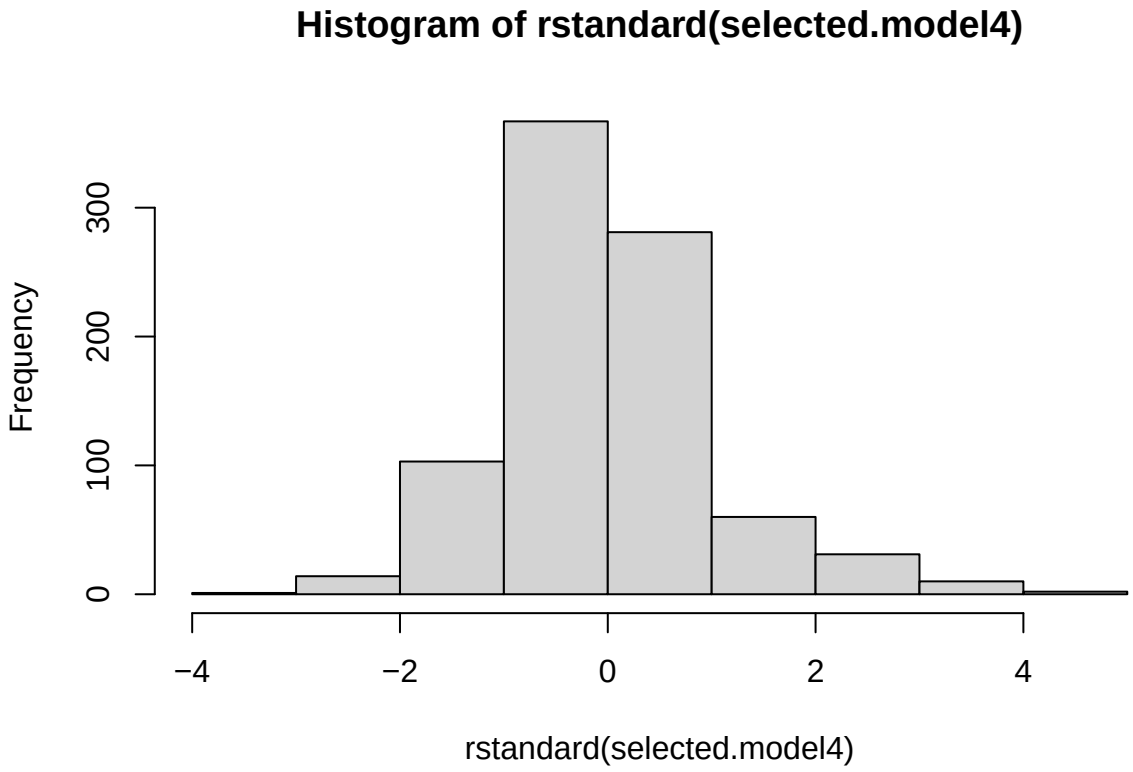

```
selected.model3 = fit.OverallQual
hist(rstandard(selected.model3))
```



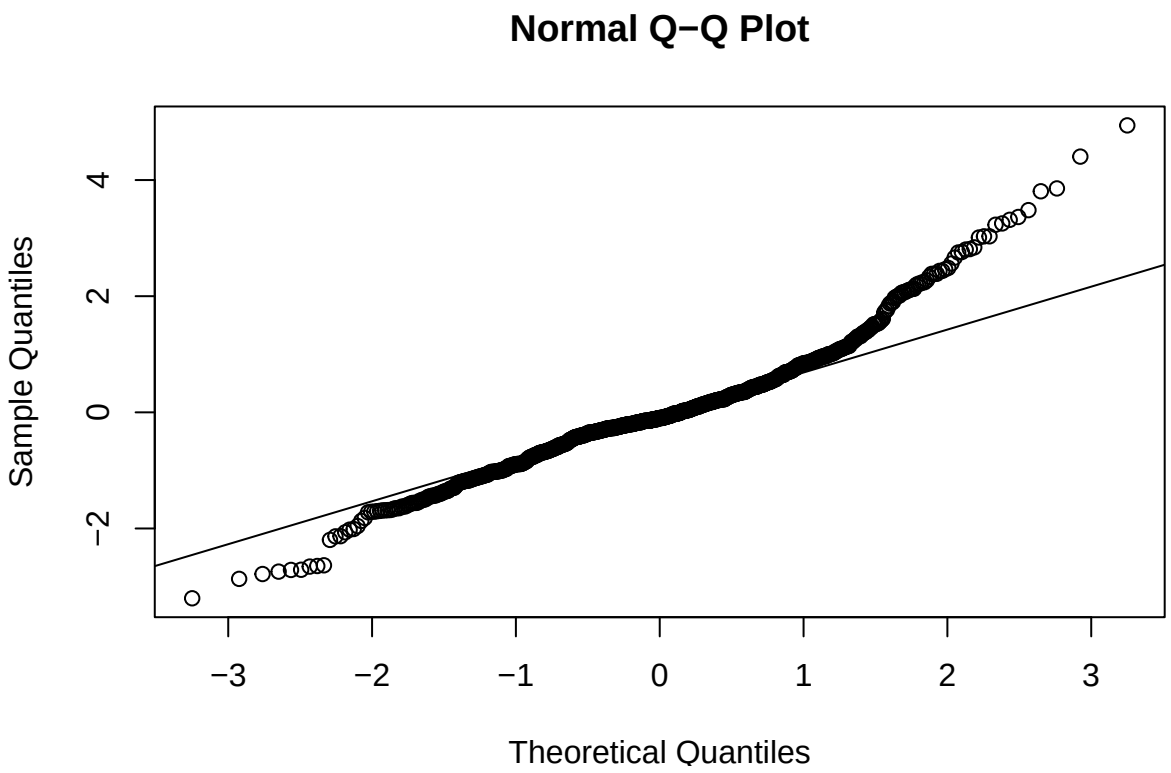

```
qqnorm(rstandard(selected.model3))
qqline(rstandard(selected.model3))
```




```
selected.model4 = fit.Neighborhood
hist(rstandard(selected.model4))
```



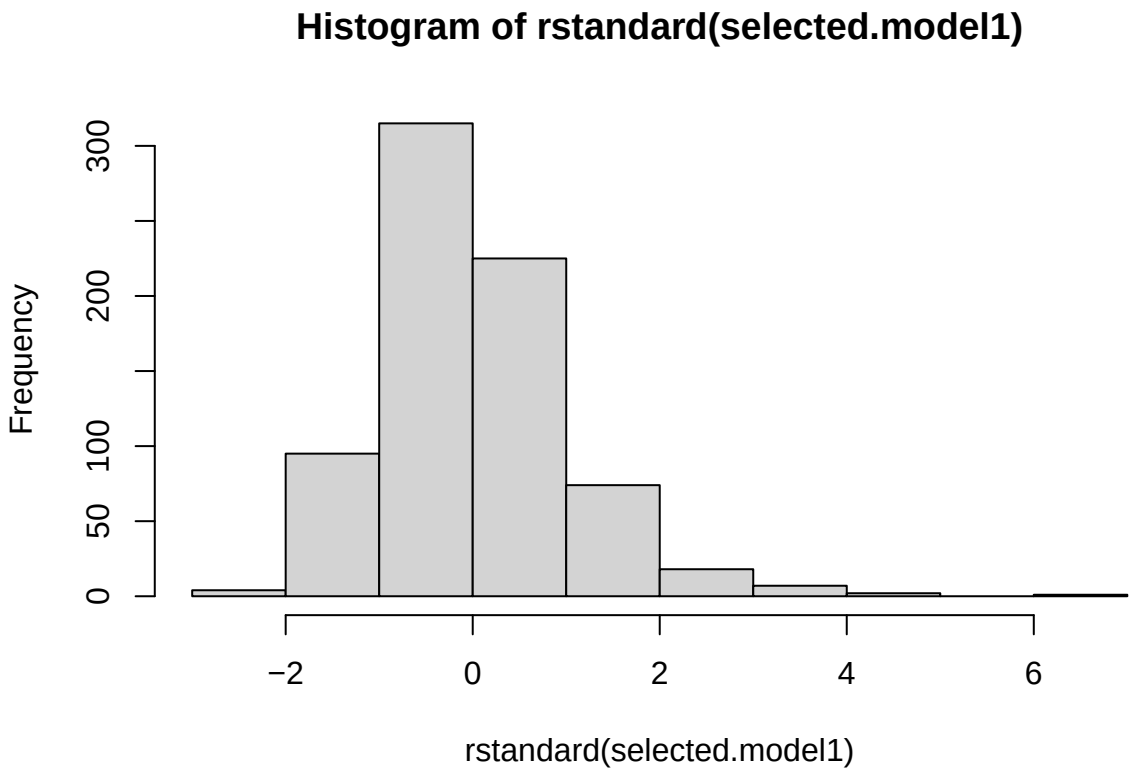

```
qqnorm(rstandard(selected.model4))
qqline(rstandard(selected.model4))
```



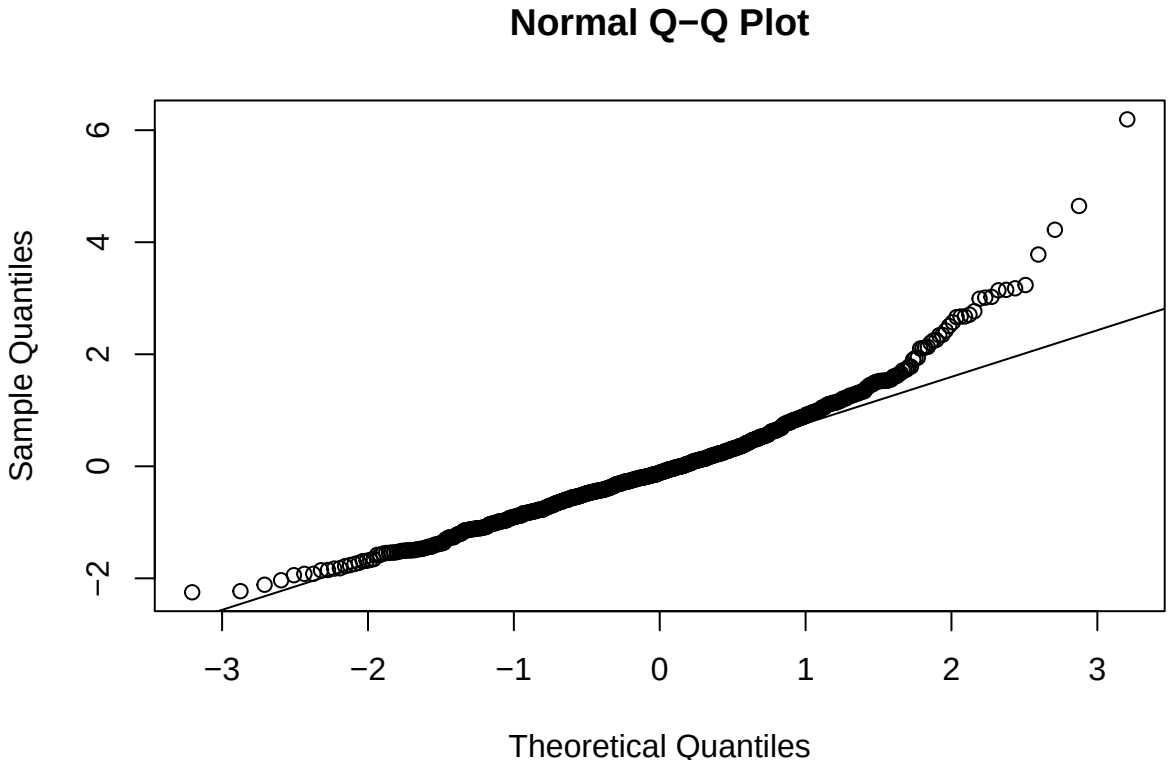

```
dataset <- dataset %>%
  group_by(YearBuilt) %>%
```

```
  mutate(Q1 = quantile(SalePrice, .25),
         Q3 = quantile(SalePrice, .75),
         IQR = IQR(SalePrice),
         lower_bound = Q1 - 1.5*IQR,
         upper_bound = Q3 + 1.5*IQR) %>%
  filter(SalePrice > lower_bound & SalePrice < upper_bound)

dataset <- dataset %>%
  group_by(Neighborhood) %>%
  mutate(Q1 = quantile(SalePrice, .25),
         Q3 = quantile(SalePrice, .75),
         IQR = IQR(SalePrice),
         lower_bound = Q1 - 1.5*IQR,
         upper_bound = Q3 + 1.5*IQR) %>%
  filter(SalePrice > lower_bound & SalePrice < upper_bound)

dataset <- dataset %>%
  group_by(OverallQual) %>%
  mutate(Q1 = quantile(SalePrice, .25),
         Q3 = quantile(SalePrice, .75),
         IQR = IQR(SalePrice),
         lower_bound = Q1 - 1.5*IQR,
         upper_bound = Q3 + 1.5*IQR) %>%
  filter(SalePrice > lower_bound & SalePrice < upper_bound)

 #Drugi nacin
# datas <- split(dataset, dataset$Neighborhood)
# data_no_outlier <- NULL
# for (i in 1:14){
#   Q1 <- quantile(datas[[i]]$SalePrice, .25)
#   Q3 <- quantile(datas[[i]]$SalePrice, .75)
#   IQR <- IQR(datas[[i]]$SalePrice)
#   Lowers <- Q1 - 1.5*IQR
#   Uppers <- Q3 + 1.5*IQR
#   out <- subset(datas[[i]], datas[[i]]$SalePrice > Lowers & datas[[i]]$SalePrice < Uppers)
#   data_no_outlier <- rbind(data_no_outlier, out)
# }
# dataset <- data_no_outlier
# boxplot(SalePrice ~ Neighborhood, data = dataset)
```

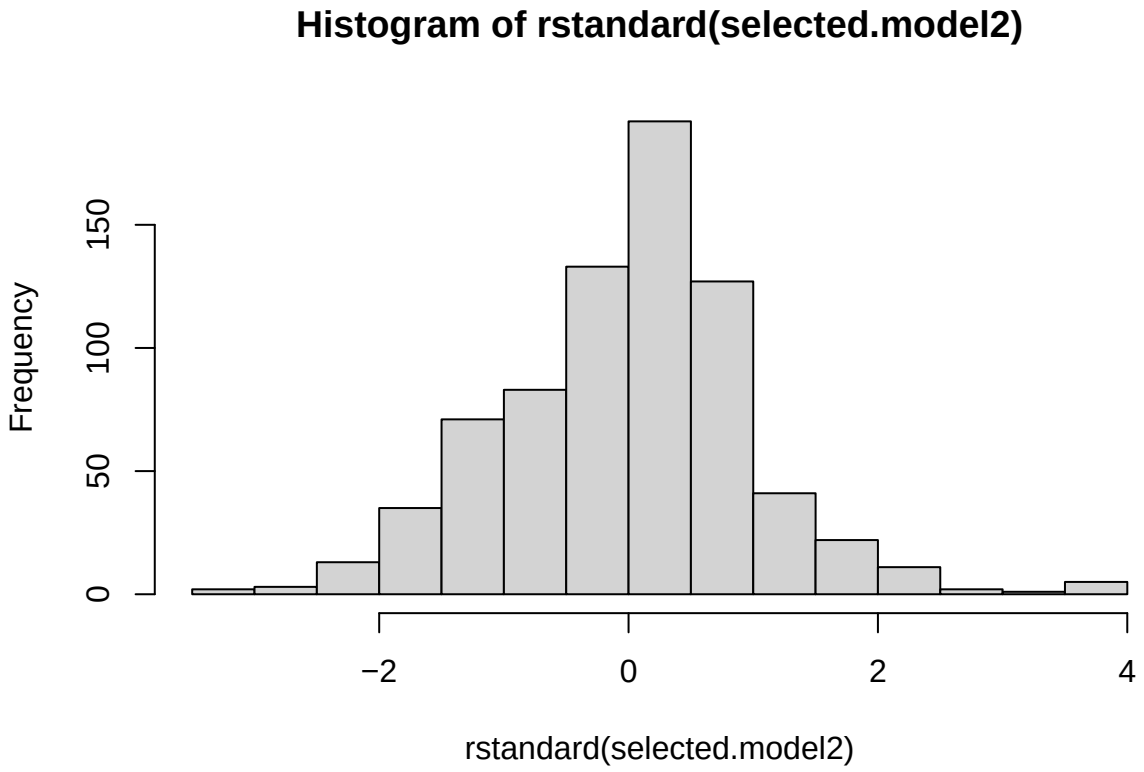
```
fit.YearBuilt = lm(SalePrice ~ YearBuilt , data=dataset)
fit.GrLivArea = lm(SalePrice ~ GrLivArea , data=dataset)
fit.OverallQual = lm(SalePrice ~ OverallQual , data=dataset)
fit.Neighborhood = lm(SalePrice ~ Neighborhood , data=dataset)

selected.model1 = fit.YearBuilt
hist(rstandard(selected.model1))
```



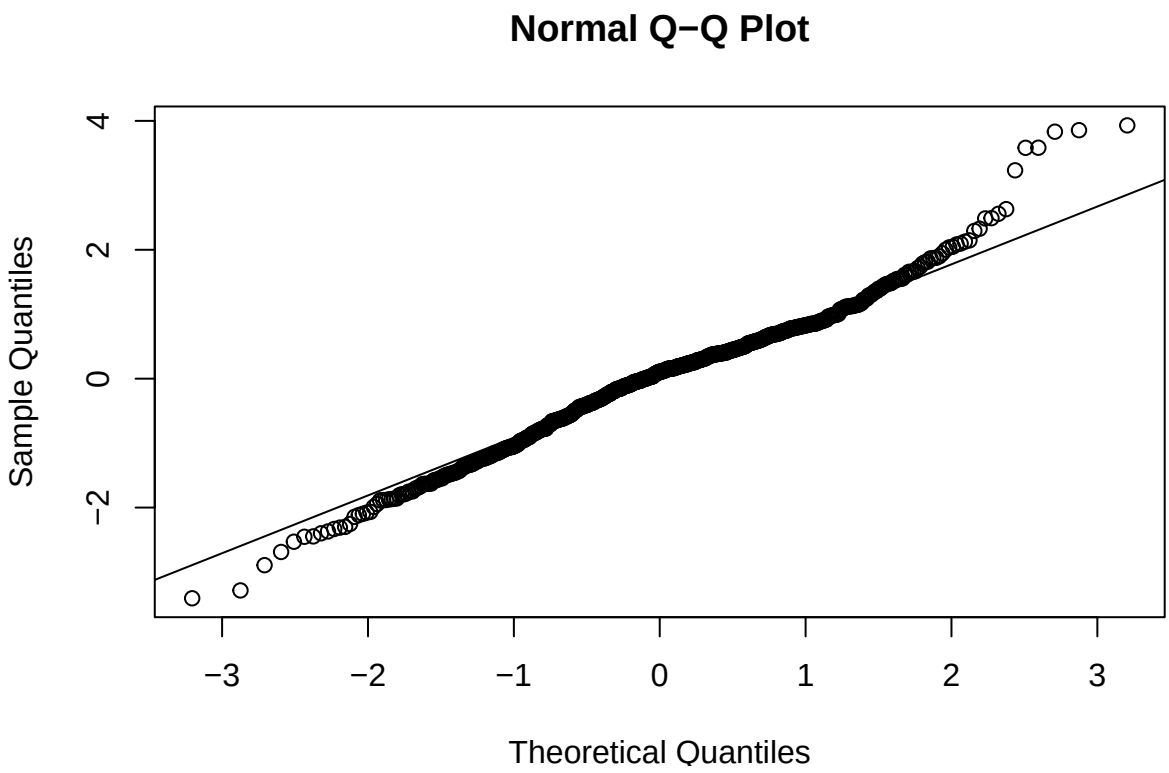

```
qqnorm(rstandard(selected.model1))
qqline(rstandard(selected.model1))
```



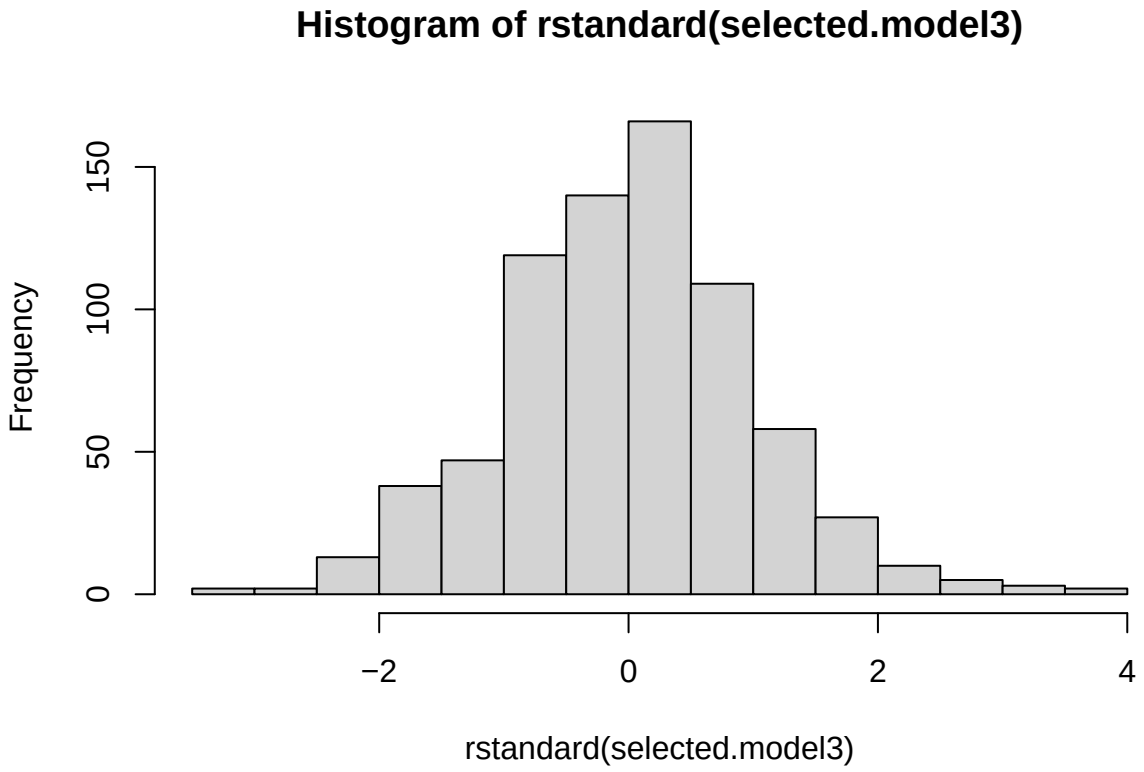

```
selected.model2 = fit.GrLivArea
hist(rstandard(selected.model2))
```



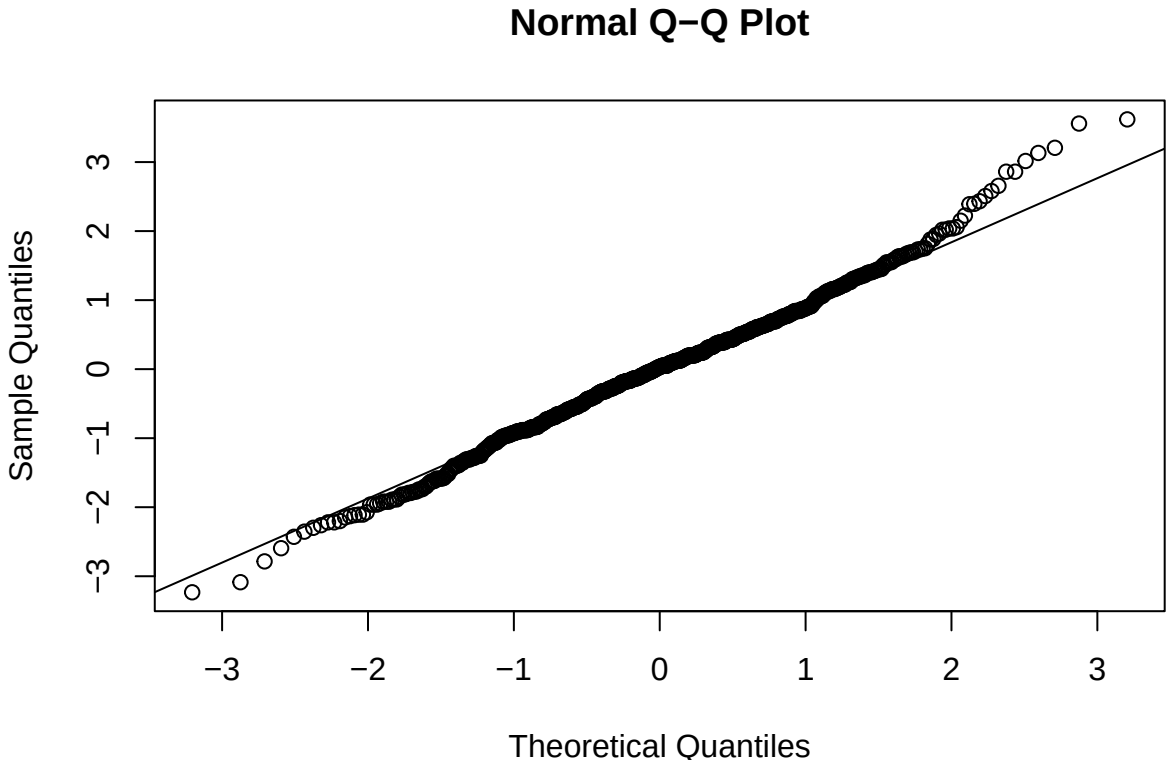

```
qqnorm(rstandard(selected.model2))
qqline(rstandard(selected.model2))
```



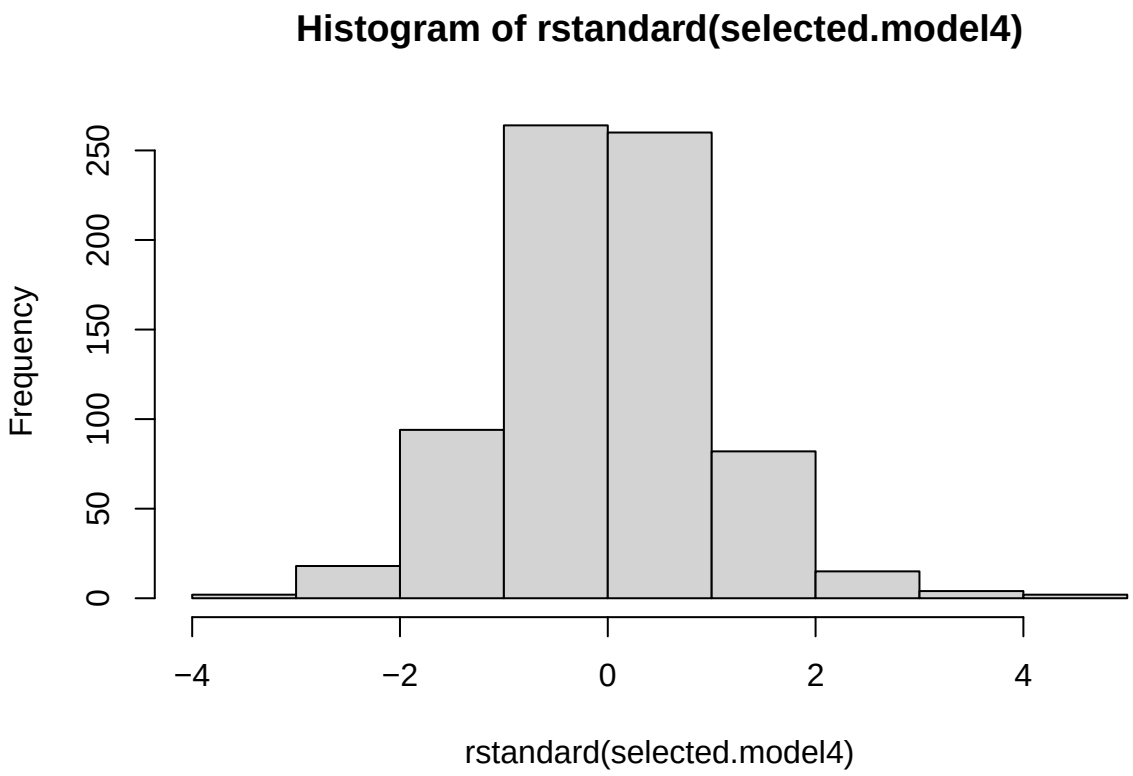

```
selected.model3 = fit.OverallQual
hist(rstandard(selected.model3))
```



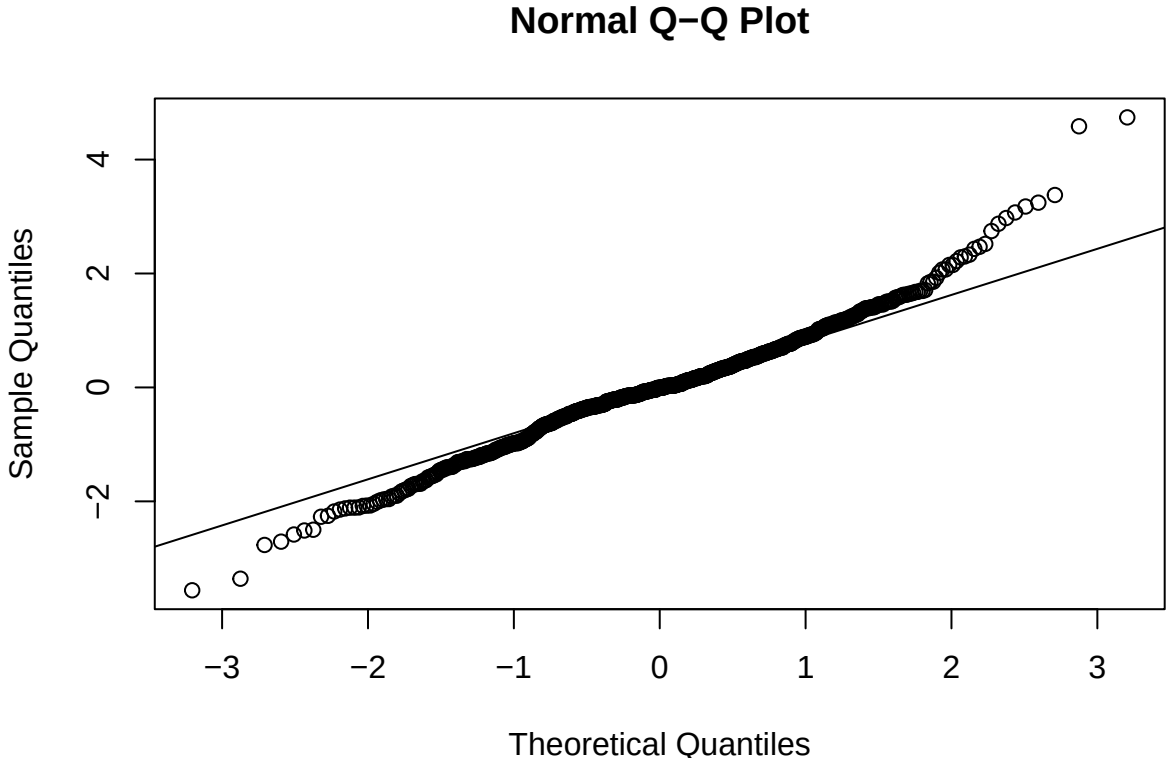

```
qqnorm(rstandard(selected.model3))
qqline(rstandard(selected.model3))
```




```
selected.model4 = fit.Neighborhood
hist(rstandard(selected.model4))
```




```
qqnorm(rstandard(selected.model4))
qqline(rstandard(selected.model4))
```




```
#require(fastDummies)
#dataset <- dummy_cols(dataset,select_columns='OverallQual')
```

```r
#dataset <- dummy_cols(dataset,select_columns='Neighborhood')

#procjena modela s dummy varijablama
fit.multi.d = lm(SalePrice ~ YearBuilt +GrLivArea + OverallQual + Neighborhood, dataset)
summary(fit.multi.d)
```

```
##
## Call:
## lm(formula = SalePrice ~ YearBuilt + GrLivArea + OverallQual +
##     Neighborhood, data = dataset)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -74367 -10348   1076  10538  91413
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -1.105e+06  1.095e+05 -10.088  < 2e-16 ***
## YearBuilt            5.756e+02  5.706e+01  10.088  < 2e-16 ***
## GrLivArea            3.964e+01  2.207e+00  17.962  < 2e-16 ***
## OverallQual          1.281e+04  9.277e+02  13.809  < 2e-16 ***
## NeighborhoodCollgCr  5.956e+03  4.860e+03   1.225   0.2208
## NeighborhoodCrawfor  2.457e+04  4.598e+03   5.345 1.21e-07 ***
## NeighborhoodEdwards -7.768e+03  3.735e+03  -2.080   0.0379 *
## NeighborhoodGilbert -1.015e+04  5.339e+03  -1.901   0.0577 .
## NeighborhoodNAmes   -4.521e+02  3.473e+03  -0.130   0.8965
## NeighborhoodNridgHt  6.615e+03  8.638e+03   0.766   0.4441
## NeighborhoodNWAmes   3.511e+03  4.492e+03   0.782   0.4347
## NeighborhoodOldTown -4.346e+03  3.539e+03  -1.228   0.2198
## NeighborhoodSawyer   3.719e+02  4.075e+03   0.091   0.9273
## NeighborhoodSawyerW  1.576e+03  4.966e+03   0.317   0.7511
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 18650 on 727 degrees of freedom
## Multiple R-squared:  0.7858, Adjusted R-squared:  0.7819
## F-statistic: 205.1 on 13 and 727 DF,  p-value: < 2.2e-16
```